

**Guía docente de la asignatura**

Asignatura	T22: DISEÑO, INTEGRACIÓN Y ADAPTACIÓN DE SOFTWARE		
Materia	TECNOLOGÍAS SOFTWARE		
Módulo			
Titulación	GRADO EN INGENIERÍA INFORMÁTICA DE SISTEMAS (464)		
Plan	464	Código	T22
Periodo de impartición	2º CUATRIMESTRE	Tipo/Carácter	OBLIGATORIA
Nivel/Ciclo	GRADO	Curso	3º
Créditos ECTS	6 ECTS		
Lengua en que se imparte	CASTELLANO		
Profesor/es responsable/s	José Manuel Marqués Corral		
Datos de contacto (E-mail, teléfono...)	TELÉFONO: 983 423000 / ext. 5638 E-MAIL: jmmc@infor.uva.es		
Horario de tutorías	Véase www.uva.es → Centros → Campus de Valladolid → Escuela Técnica Superior de Ingeniería Informática → Tutorías		
Departamento	Departamento de Informática.		



1. Situación / Sentido de la Asignatura

El diseño de software desempeña un papel fundamental en el desarrollo del software¹ y constituye una de las principales áreas de conocimiento de la ingeniería del software. El diseño de software incluye el conjunto de conceptos, técnicas, estrategias, representaciones y patrones utilizados para determinar los componentes o sistemas a implementar.

La propia evolución y complejidad de los actuales productos y sistemas software ha llevado a la necesidad de considerar los sistemas desde una perspectiva arquitectónica. En la actualidad los productos y sistemas software están formados por un gran número de componentes heterogéneos que interactúan en muchas ocasiones en entornos distribuidos. La experiencia está demostrando que partir de un buen diseño la arquitectura puede reducir drásticamente las tasas de fallo.

En esta asignatura se proporcionará a los estudiantes una introducción a los principios y las metodologías de diseño de las arquitecturas software de mayor difusión y aceptación tanto en el ámbito académico como en el industrial. La asignatura se divide en dos grandes bloques, que se corresponden con los diferentes niveles en los que se pueden clasificar las arquitecturas de las tecnologías de la información: arquitectura empresarial (*Enterprise Architecture*) y arquitectura de aplicación (*Application Architecture*). Es cierto que puede considerarse, algunos autores y soluciones concretas así lo hacen, un nivel intermedio, la arquitectura de proyecto. Sin embargo, dado el nivel al que se imparte la asignatura, la formación previa de los alumnos y, sobre todo, las limitaciones temporales, se considera que el estudio de los dos niveles propuestos es suficiente. En base a estas últimas observaciones, se ha estimado que el mayor énfasis ha de ponerse en la formación relacionada con el nivel de arquitectura de aplicación software, dejando la formación en arquitecturas de empresa en un nivel muy introductorio y elemental.

Para la formación en arquitecturas de aplicación, la asignatura se centrará en el estudio de los fundamentos teóricos y prácticos básicos para el modelado y documentación del diseño de alto nivel de un sistema. Se persigue que el estudiante comprenda el concepto de arquitectura software, que conozca alguno de los patrones y frameworks arquitectónicos de mayor utilización y difusión y, por supuesto, que sea capaz de desarrollar y documentar el diseño arquitectónico y detallado de un sistema software.

La formación en diseño y arquitecturas de aplicación será complementada con una introducción a las arquitecturas empresariales y el modelado de procesos de negocio a fin de cubrir, aunque sea mínimamente, aquellos aspectos relacionados con la visión global y la estructura de las organizaciones y su alineamiento con las tecnologías de la información que posibilitan su funcionamiento.

1.1 Contextualización

Esta asignatura forma parte de la materia "Tecnologías Software" encuadrado en el bloque "Tecnología específica- Tecnologías de la Información". La asignatura está claramente orientada a conseguir que los

¹ Software Engineering Body of Knowledge (SWEBOK), Bourque, P. & Dupuis, R. (ed.). IEEE Computer Society, 2004



alumnos sean capaces de adquirir las habilidades necesarias para abordar el diseño de un sistema informático aplicando principios, métodos, herramientas y prácticas propias de la ingeniería.

1.2 Relación con otras materias

Fundamentos de Ingeniería del Software, Sistemas distribuidos, Fundamentos de Programación, Programación Orientada a la Integración de Sistemas, Interacción Persona – Computador, Diseño de Bases de Datos.

1.3 Prerrequisitos

Se recomienda que los alumnos hayan cursado con aprovechamiento al menos las siguientes asignaturas del vigente plan de estudios: Fundamentos de Ingeniería del Software, Sistemas Distribuidos, Interacción Persona-Computadora y Diseño de Bases de Datos.

Se recomienda que el alumno tenga un nivel de inglés que le permita la comprensión de bibliografía básica y complementaria recomendada para la asignatura.

Dadas las características inherentes a todo proyecto de diseño de un sistema software, son muy recomendables la constancia, la iniciativa personal y la predisposición al trabajo colativo en grupo. La asimilación de los contenidos teóricos de la asignatura implica, por una parte, la capacidad de lectura crítica de los textos básicos y complementarios puestos a su disposición, pero además la búsqueda proactiva del material complementario en la red publicado por los grandes fabricantes de hardware y software, y que resulta imprescindible para el desarrollo de un buen diseño.



2. Competencias

2.1 Generales

Código	Descripción
G3.	Capacidad de análisis y síntesis
G5.	Comunicación oral y escrita en la lengua propia
G6.	Conocimiento de una segunda lengua (preferentemente inglés)
G8.	Habilidades de gestión de la información
G9.	Resolución de problemas
G10.	Toma de decisiones
G11.	Capacidad crítica y autocrítica
G12.	Trabajo en equipo
G16.	Capacidad de aplicar los conocimientos en la práctica
G17.	Habilidades de investigación
G18.	Capacidad de aprender
G19.	Capacidad de adaptarse a nuevas situaciones
G20.	Capacidad de generar nuevas ideas
G21.	Habilidad para trabajar de forma autónoma

2.2 Específicas

TI1.	Capacidad para comprender el entorno de una organización y sus necesidades en el ámbito de las tecnologías de la información y las comunicaciones.
TI2.	Capacidad para seleccionar, diseñar, desplegar, integrar, evaluar, construir, gestionar, explotar y mantener las tecnologías de hardware, software y redes, dentro de los parámetros de coste y calidad adecuados.
TI3.	Capacidad para emplear metodologías centradas en el usuario y la organización para el desarrollo, evaluación y gestión de aplicaciones y sistemas basados en tecnologías de la información que aseguren la accesibilidad, ergonomía y usabilidad de los sistemas.
TI5.	Capacidad para seleccionar, desplegar, integrar y gestionar sistemas de información que satisfagan las necesidades de la organización, con los criterios de coste y calidad identificados.
TI6.	Capacidad de concebir sistemas, aplicaciones y servicios basados en tecnologías de red, incluyendo Internet, web, comercio electrónico, multimedia, servicios interactivos y computación móvil.
SI1.	Capacidad de integrar soluciones de Tecnologías de la Información y las Comunicaciones y procesos empresariales para satisfacer las necesidades de información de las organizaciones, permitiéndoles alcanzar sus objetivos de forma efectiva y eficiente, dándoles así ventajas competitivas.
SI3.	Capacidad para participar activamente en la especificación, diseño, implementación y mantenimiento de los sistemas de información y comunicación.



3. Objetivos

Código	Descripción
TI1.1	Evaluar las alternativas de diseño frente a integración software.
TI2.1	Conocer la arquitectura de distintos tipos de sistemas: centralizados, distribuidos, heterogéneos
TI3.1	Aplicar los patrones de diseño adecuados para el desarrollo e integración de software

4. Tabla de dedicación del estudiante a la asignatura

ACTIVIDADES PRESENCIALES	HORAS	ACTIVIDADES NO PRESENCIALES	HORAS
Clases teórico-prácticas (T/M)	30	Estudio y trabajo autónomo individual	45
Clases prácticas de aula (A)		Estudio y trabajo autónomo grupal	45
Laboratorios (L)	28		
Prácticas externas, clínicas o de campo			
Seminarios (S)	2		
Tutorías grupales (TG)			
Evaluación (fuera del periodo oficial de exámenes)	0		
Total presencial	60	Total no presencial	90

5. Bloques temáticos

Bloque 1: Arquitecturas y diseño de aplicaciones software.

Carga de trabajo en créditos ECTS:

a. Contextualización y justificación

La mayoría de los estándares y recomendaciones curriculares relacionados con la enseñanza universitaria en ingeniería del software establecen la arquitectura y el diseño software como troncales. El *Computing Curriculum Software Engineering*² recomienda el diseño y la arquitectura software como una de sus diez áreas esenciales. El ACM Computing Curricula 2004 recomienda el diseño software tanto para la especialización en ingeniería del software como en la especialización en Computer Science.

En este bloque se estudiarán aquellos aspectos más relevantes y de mayor difusión en la industria del software de la disciplina de la arquitectura y el diseño software. Se trabajará, desde un enfoque coherente e integrado, en las técnicas, métodos y herramientas relacionadas con el diseño de alto nivel de aplicaciones software.

Los contenidos de este bloque se estructuran de la siguiente manera: en primer lugar se presentarán los conceptos, objetivos y definiciones del diseño, para a continuación introducir el concepto de arquitectura del software, los estilos arquitectónicos y las estructuras genéricas para los tipos de aplicaciones más habituales.

² <http://sites.computer.org/ccse/>



Por último se aborda el modelado y descripción del diseño conceptual y técnico desde una perspectiva de desarrollo soportado por patrones, utilizando los métodos del diseño orientado al objeto y el lenguaje de modelado UML 2.0 para la documentación de los diseños elaborados.

b. Objetivos de aprendizaje

Código	Descripción
TI1.1	Evaluar las alternativas de diseño frente a integración software.
TI3.1	Aplicar los patrones de diseño adecuados para el desarrollo e integración de software

c. Contenidos

Programa de Teoría

1. Introducción a la arquitectura y el diseño de software.
2. Diseño arquitectónico software. Visión general
3. Diseño de la arquitectura software. Patrones arquitectónicos.
4. Diseño de software. Métodos y técnicas.

d. Métodos docentes

Ver anexo: Métodos docentes.

Plan de trabajo

e.

Ver anexo: Cronograma

f. Evaluación

Ver apartado 7. Sistema de calificaciones.

g. Bibliografía básica

- [1] F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal, y M. Stal, *Pattern-Oriented Software Architecture Volume 1: A System of Patterns*, Volume 1. Wiley, 1996.
- [2] M. Fowler, *Patterns of Enterprise Application Architecture*, 1.ª ed. Addison-Wesley Professional, 2002
- [3] C. Larman, *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, 3rd Edition*, Dimensions: 8x10. 2004.
- [4] I. Sommerville, *Ingeniería Del Software*. Pearson Educación, 2005.

h. Bibliografía complementaria

- L. Bass, P. Clements, y R. Kazman, *Software Architecture in Practice*, 2.ª ed. Addison-Wesley Professional, 2003.
- H. Goma, *Software Modeling and Design: UML, Use Cases, Patterns, and Software Architectures*, Cambridge University Press, 2011.
- R. N. Taylor, N. Medvidovic, y E. M. Dashofy, *Software Architecture: Foundations, Theory, and Practice*, 1.ª ed. Wiley, 2009.



i. Recursos necesarios

- » «Microsoft Application Architecture Guide, 2nd Edition». [Online]. Available: <http://msdn.microsoft.com/en-us/library/ff650706.aspx>. [Accessed: 31-may-2012].
- » «OpenUP». [Online]. Available: <http://epf.eclipse.org/wikis/openup/>. [Accessed: 31-may-2012].
- » «Software Architecture | Overview». [Online]. Available: <http://www.sei.cmu.edu/architecture/?location=secondary-nav&source=18823>. [Accessed: 30-may-2012].
- » «What is a software architecture?» [Online]. Available: <http://www.ibm.com/developerworks/rational/library/feb06/eeles/>. [Accessed: 30-may-2012].

Bloque 2: Arquitectura e integración de aplicaciones de empresa.

Carga de trabajo en créditos ECTS:

a. Contextualización y justificación

Este bloque se centra en el papel que la Arquitectura Empresarial (EA) -- *Enterprise Architecture* en la terminología inglesa – desempeña en una gestión integrada e integral de las tecnologías de la información en las organizaciones. Se persigue que el alumno conozca las relación entre la EA y la infraestructura de información de una organización y en cómo las EA condicionan y dirigen el desarrollo de las soluciones concretas en tecnologías de la información (TI).

En primer lugar, se presentan los conceptos básicos y una visión general de las prácticas actuales en el campo de las arquitectura de empresa. A continuación, se introduce a alumno en algunas de las técnicas, métodos y herramientas de mayor aceptación y difusión relacionadas con el modelado de las arquitecturas empresariales y de los procesos de negocio. El bloque finaliza con una somera exposición de productos y tecnologías que posibilitan la integración de los sistemas software solución en las organizaciones.

b. Objetivos de aprendizaje

Código	Descripción
TI1.1	Evaluar las alternativas de diseño frente a integración software.
TI3.1	Aplicar los patrones de diseño adecuados para el desarrollo e integración de software

c. Contenidos

Programa de Teoría

1. Arquitectura e integración de aplicaciones de empresa.

d. Métodos docentes

Ver anexo: Métodos docentes.

e. Plan de trabajo

Ver anexo: Cronograma.



f. Evaluación

Ver apartado 7. Sistema de calificaciones.

g. Bibliografía básica

M. Lankhorst, *Enterprise Architecture at Work: Modelling, Communication and Analysis*, 2nd ed. Springer, 2009.

h. Bibliografía complementaria

i. Recursos necesarios

- » «BPMN Information Home». [Online]. Available: <http://www.bpmn.org/>. [Accessed: 01-jun-2012].
- » «Enterprise Architecture». [Online]. Available: <http://msdn.microsoft.com/es-es/architecture/bb469938.aspx>. [Accessed: 29-may-2012].
- » «TOGAF® Version 9.1». [Online]. Available: <http://www.opengroup.org/togaf/>. [Accessed: 01-jun-2012].

6. Temporalización (por bloques temáticos)

BLOQUE TEMÁTICO	CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
Arquitecturas y diseño de aplicaciones software	5 ECTS	Semanas 1 a 12
Integración y arquitectura de aplicaciones de empresa	1 ECTS	Semanas 13 a 15

7. Sistema de calificaciones – Tabla resumen

INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Trabajos evaluables de laboratorio		Fecha aproximada de realización:
1. Modelado de la arquitectura del sistema. D. Paquetes, D. Componentes y D. Despliegue	5%	1. Semanas 1 a 4
2. Modelado de los aspectos dinámicos del sistema. D. Actividad	5%	2. Semanas 5 a 6
3. Especificación de la arquitectura software.	5%	3. Semanas 7 a 8
4. Diseño de la capa de lógica de negocio	5%	4. Semanas 9 a 10
5. Diseño de la capa de acceso a datos	5%	5. Semanas 11 a 12
6. Diseño con patrones GoF.	5%	6. Semanas 13 a 15
Examen final escrito	70%	Periodo de exámenes

CRITERIOS DE CALIFICACIÓN

- **Convocatorias ordinaria y extraordinaria:**
 - El examen de la parte teórica se realizará mediante examen escrito sobre las materias incluidas en el programa de la asignatura. Incluirá la elaboración de modelos de diseño correspondientes a supuestos prácticos. Se valorará el ajuste de la solución elaborada al problema planteado y la correcta utilización de los principios y notaciones de diseño.
 - La evaluación de la parte práctica se realizará sobre el trabajo de laboratorio realizado en grupo a lo largo del curso.
 - Para poder superar la asignatura será necesario tener superadas por separado cada una de las partes de la asignatura, la teórica y la práctica.
 - La parte de laboratorio se supera obteniendo más de un 1,5 puntos sobre los 3 puntos posibles a obtener en el conjunto de trabajos evaluables y habiendo obtenido puntuación estrictamente mayor que cero en cada uno de los trabajos evaluables.
 - La parte de teoría se supera obteniendo más de 3 puntos sobre los 7 puntos posibles a obtener en el examen final escrito.
 - La nota final será la suma de las calificaciones obtenidas en cada una de las partes en las que se divide el sistema de calificaciones, siempre que se superen individualmente cada una de las partes de acuerdo con los criterios anteriormente expuestos.

8. Anexo: Métodos docentes

Actividad	Metodología
Clase de teoría	<ul style="list-style-type: none"> • Clase magistral participativa • Estudio de casos en aula • Resolución de problemas
Clase práctica	<ul style="list-style-type: none"> • Realización de ejercicios y pequeños supuestos prácticos. El trabajo se realizará en grupos (2/3 alumnos), siguiendo un enfoque colaborativo.
Seminarios	<ul style="list-style-type: none"> • Talleres de aprendizaje
Tutoría	<ul style="list-style-type: none"> • Evaluación de los contenidos teóricos y de los proyectos

9. Anexo: Cronograma de actividades previstas

Semana	Teoría	Seminario	Laboratorio	Contenidos /Actividades	
1	2	2		2T	Introducción. Conceptos básicos.
				2S	UML. Herramienta CASE
2	2		2	2T	Introducción. Proceso y patrones de diseño
				2L	Modelado de la arquitectura del sistema. D. Paquetes
3	2		2	2T	Diseño arquitectónico. Visión general
				2L	Modelado de la arquitectura del sistema. D. Componentes
4	2		2	2T	Organización y descomposición de los sistemas software.
				2L	Modelado de la arquitectura del sistema. D. Despliegue
5	2		2	2T	Diseño de la arquitectura. Estructuración del sistema. POSA (I)
				2L	Modelado de los aspectos dinámicos del sistema. D. Actividad
6	2		2	2T	Diseño de la arquitectura. Estructuración del sistema. POSA (II)
				2L	Modelado de los aspectos dinámicos del sistema. D. Actividad
7	2		2	2T	Diseño de la arquitectura lógica del sistema. Subsistemas.
				2L	Especificación de la arquitectura software
8	2		2	2T	Diseño de la arquitectura. Capa Interacción con el usuario.
				2L	Especificación de la arquitectura software
9	2		2	2T	Diseño de la arquitectura. Capa acceso a datos.
				2L	Diseño capa Lógica de Dominio (lógica del negocio)
10	2		2	2T	Diseño de la arquitectura. Integración de aplicaciones.
				2L	Diseño capa Lógica de Dominio (lógica del negocio)



Semana	Teoría	Seminario	Laboratorio	Contenidos /Actividades	
11	2		2	2T	Diseño de software. OOD
				2L	Diseño capa de acceso a datos
12	2		2	2T	Diseño de software. Diseño con patrones GoF (I)
				2L	Diseño capa acceso a datos
13	2		2	2T	Diseño de software. Diseño con patrones GoF (II)
				2L	Diseño con patrones GoF
14	2		2	2T	Introducción a las Arquitecturas de Empresa.
				2L	Diseño con patrones GoF
15	2		2	2T	Modelado de los procesos de negocio.
				2L	Diseño con patrones GoF

